CLAIMS

None of the claims has been amended. The claims are reproduced here for the Examiner's convenience.

1. (Original) A method in a computer system for restricting network address-based communication by selected processes to a set of specific network addresses, the method comprising:

associating at least one selected process with at least one network address;

determining whether an attempted network address-based communication of a selected process is via an associated address; and

in response to a determination that the communication is via an associated address, allowing the communication to proceed.

2. (Original) The method of claim 1 further comprising:

loading at least one selected process into computer memory; and

storing at least one association, between the process and at least one network address.

3. (Original) The method of claim 1 wherein:

associations between selected processes and network addresses are stored in an association table in a computer memory of the computer system.

4. (Original) The method of claim 3 wherein:

the association table is stored in operating system address space.

5. (Original) The method of claim 1 wherein:

a network address-based communication comprises an attempt to designate a network address to be used for subsequent communication.

6. (Original) The method of claim 1 wherein:

a network address-based communication comprises an attempt to associate a communication channel with a network address.

7. (Withdrawn) The method of claim 1 wherein:

a network address-based communication comprises an attempt to communicate without designating a network address to be used for communication.

8. (Original) The method of claim 1 wherein:

a network address-based communication comprises an attempt to establish a connection to a second process.

9. (Original) The method of claim 1 wherein:

a network address-based communication comprises an attempt to transmit data to a second process.

10. (Original) The method of claim 9 wherein:

the second process is executing in a computer memory of the computer system.

11. (Original) The method of claim 9 wherein:

the second process is executing in a computer memory of a second computer system.

12. (Original) The method of claim 1 further comprising:

determining whether an attempted network address-based communication is via an associated address by intercepting system calls that pertain to network address-based communication.

13. (Original) The method of claim 12 further comprising:

storing object code that determines whether an attempted network address-based communication is via an associated network address; and

wherein intercepting comprises replacing a pointer to a system call with a pointer to the object code, such that calling the system call causes the object code to execute.

14. (Original) The method of claim 13 further comprising:

loading an interception module into computer memory, the interception module comprising the object code.

15. (Original) The method of claim 14 wherein:

the interception module is loaded into a running operating system kernel.

16. (Original) The method of claim 13 wherein determining whether an attempted network address-based communication is via an associated network address comprises:

examining at least one stored association to determine whether the processes that called the system call is associated with at least one network address; and

in response to a determination that the processes is associated with at least one network address, determining whether the attempted communication is via an associated network address.

17. (Original) The method of claim 1 further comprising:

determining whether an attempted network address-based communication is via an associated address by modifying a communication protocol stack so as to intercept communication protocol subroutines that pertain to network address-based communication.

18. (Original) The method of claim 17 further comprising:

storing object code that determines whether an attempted network address-based communication is via an associated network address; and

wherein intercepting comprises replacing a pointer to a subroutine with a pointer to the object code, such that calling the subroutine call causes the object code to execute.

19. (Original) The method of claim 18 further comprising:

loading an interception module into computer memory, the interception module comprising the object code.

20. (Original) The method of claim 19 wherein:

the interception module is loaded into a running operating system kernel.

21. (Original) The method of claim 18 wherein determining whether an attempted network address-based communication is via an associated network address comprises:

examining at least one stored association to determine whether the process that called the subroutine is associated with at least one network address; and

in response to a determination that the processes is associated with at least one network address, determining whether the attempted communication is via an associated network address.

22. (Original) The method of claim 17 wherein:

the communication protocol stack that is modified is a Transmission Control Protocol/Internet Protocol stack.

23. (Original) The method of claim 1 further comprising:

detecting creation of a child process by a selected process;

associating the child process with all network addresses with which the selected process is associated.

24. (Original) The method of claim 23 further comprising:

detecting creation of a child process by intercepting system calls that create child processes.

25. (Original) The method of claim 24 further comprising:

storing object code that detects creation of a child process by a selected process, and that associates the child process with all network addresses with which the selected process is associated; and

wherein intercepting comprises replacing a pointer to a system call with a pointer to the object code, such that calling the system call causes the object code to execute.

26. (Original) The method of claim 25 further comprising:

loading an interception module into computer memory, the interception module comprising the object code.

27. (Original) The method of claim 26 wherein:

the interception module is loaded into a running operating system kernel.

28. (Original) The method of claim 25 wherein associating comprises:

storing an association between the child processes and a network address.

29. (Original) The method of claim 1 further comprising:

associating a child process of a selected process with a single network address with which the selected process is associated;

determining whether network address-based communication of the child process is via the associated address; and

in response to a determination that the communication is via the associated address, allowing the communication to proceed.

30. (Original) The method of claim 1 further comprising:

associating a child process of a selected process with at least two network addresses with which the selected process is associated;

determining whether network address-based communication of the child process is via an associated address; and

in response to a determination that the communication is via an associated address, allowing the communication to proceed.

31. (Original) The method of claim 1 further comprising:

detecting termination of a selected process; and

deleting all associations between the process and network addresses.

32. (Original) The method of claim 31 further comprising:

detecting termination of a selected process by intercepting system calls that terminate processes.

33. (Original) The method of claim 32 further comprising:

storing object code that deletes all associations between a selected process and network addresses; and

wherein intercepting comprises replacing a pointer to a system call with a pointer to the object code, such that calling the system call causes the object code to execute.

34. (Original) The method of claim 33 further comprising:

loading an interception module into computer memory, the interception module comprising the object code.

35. (Original) The method of claim 34 wherein:

the interception module is loaded into a running operating system kernel.

36. (Original) The method of claim 31 wherein deleting comprises:

deleting all associations between a selected process and network addresses.

37. (Original) The method of claim 1 further comprising:

in response to a determination that the attempted communication is not via an associated network address, generating an error condition.

38. (Original) The method of claim 37 wherein:

generating an error condition comprises returning an error code.

39. (Original) The method of claim 37 wherein:

generating an error condition comprises throwing an exception.

40. (Original) The method of claim 37 further comprising:

in response to generating an error condition, not allowing the communication to proceed.

41. (Original) The method of claim 1 wherein the set consists of one network address.

42. (Original) The method of claim 1 wherein the set consists of at least two network addresses.

43. (Original) A method in a computer system for restricting network address-based communication by selected processes to a set of specific network addresses, the method comprising:

associating at least one selected process with at least one network address;

determining whether an attempted network address-based communication of a selected process is via an associated address; and

in response to a determination that the attempted communication is not via an associated address, not allowing the attempted communication to proceed.

44. (Original) A method in a computer system for restricting network address-based communication by selected processes to specific network addresses, the method comprising:

associating at least one selected process with at least one network address;

detecting an attempt by a selected processes to associate a communication channel with a network address; and

determining whether the network address with which the selected process is attempting to associate a communication channel is associated with the selected process.

45. (Original) The method of claim 44 further comprising:

in response to a determination that the network address is associated with the selected process, allowing the communication channel to be associated with the network address.

46. (Original) The method of claim 44 further comprising:

in response to a determination that the network address is not associated with the selected process, not allowing the communication channel to be associated with the network address.

47. (Original) A method in a computer system for restricting network address-based communication by selected processes to specific network addresses, the method comprising:

associating at least one selected process with at least one network address;

detecting an attempt by a selected processes to associate a communication channel with a network address, wherein a provided value for the network address comprises a wild card; and

associating the communication channel with a network address that is associated with the process.

48. (Original) The method of claim 47 wherein:

the selected process is associated with a single network address; and

associating the communication channel with the single network address.

49. (Original) The method of claim 47 wherein the selected process is associated with multiple network addresses; the method further comprising:

associating the communication channel with one of the multiple network addresses, resulting in a communication channel-network address pair;

establishing one communication channel per each additional one of the multiple network addresses;

associating each established communication channel with one of the multiple network addresses, resulting in additional communication channel-network address pairs; and

associating the communication channel with the communication channel, network address pairs.

50. (Original) A method in a computer system for restricting network address-based communication by selected processes to specific network addresses, the method comprising:

associating at least one selected process with a unique local host address;

detecting an attempt by a selected process to communicate with a local host; and

designating the unique local host address associated with the selected process to be used by the selected process to communicate with the local host.

51. (Withdrawn) A method in a computer system for restricting network address-based communication by selected processes to specific network addresses, the method comprising:

associating at least one selected process with at least one network address;

detecting an attempt by a selected process to communicate with a second process via a communication channel;

determining if the communication channel is associated with a network address; and

in response to determining that the communication channel is not associated with a network address, associating the communication channel with a network address that is associated with the process.

52. (Withdrawn) The method of claim 51 further comprising:

in response to a determination that the communication channel is associated with a network address that is associated with the selected process, allowing subsequent communication via the communication channel.

53. (Withdrawn) The method of claim 51 further comprising:

in response to a determination that the communication channel is associated with a network address that is not associated with the selected process, not allowing subsequent communication via the communication channel.

54. (Withdrawn) A method in a computer system for restricting network address-based communication by selected processes to specific network addresses, the method comprising:

associating at least one selected process with at least one network address;

detecting an attempt by a selected process to establish a connection between a communication channel and a second process;

determining if the communication channel is associated with a network address; and

in response to determining that the communication channel is not associated with a network address, associating the communication channel with a network address that is associated with the selected process.

55. (Withdrawn) The method of claim 54 further comprising:

in response to a determination that the communication channel is associated with a network address that is associated with the selected process, allowing the connection to be established.

56. (Withdrawn) The method of claim 54 further comprising:

in response to a determination that the communication channel is associated with a network address that is not associated with the selected process, not allowing the connection to be established.

57. (Cancelled)

58. (Original) A computer program product for restricting network address-based communication by selected processes to a set of specific network addresses, the computer program product comprising:

program code for associating at least one selected process with at least one network address;

program code for determining whether an attempted network address-based communication of a selected process is via an associated address;

program code for, in response to a determination that the communication is via an associated address, allowing the communication to proceed; and

a computer readable medium on which the program codes are stored.

59. (Original) The computer program product of claim 58 further comprising:

program code for loading at least one selected process into computer memory; and

program code for storing at least one association between the process and at least one network address.

60. (Original) The computer program product of claim 58 further comprising:

program code for determining whether an attempted network address-based communication is via an associated address by intercepting system calls that pertain to network address-based communication.

61. (Original) The computer program product of claim 58 further comprising:

program code for determining whether an attempted network address-based communication is via an associated address by modifying a communication protocol stack so as to intercept communication protocol subroutines that pertain to network address-based communication.

62. (Original) The computer program product of claim 61 further comprising:

program code for storing object code that determines whether an attempted network address-based communication is via an associated network address; and

program code for replacing a pointer to a subroutine with a pointer to the object code, such that calling the subroutine call causes the object code to execute.

63. (Original) The computer program product of claim 62 further comprising:

program code for loading an interception module into computer memory, the interception module comprising the object code.

64. (Original) The computer program product of claim 62 further comprising:

program code for examining at least one stored association to determine whether the processes that called the subroutine is associated with at least one network address; and

program code for, in response to a determination that the processes is associated with at least one network address, determining whether the attempted communication is via an associated network address.

65. (Original) The computer program product of claim 58 further comprising:

program code for detecting creation of a child process by a selected process; and

program code for associating the child process with all network addresses with which the selected process is associated.

66. (Original) The computer program product of claim 65 further comprising:

program code for detecting creation of a child process by intercepting system calls that create child processes.

67. (Original) The computer program product of claim 66 further comprising:

program code for storing object code that detects creation of a child process by a selected process, and that associates the child process with all network addresses with which the selected process is associated; and

program code for replacing a pointer to a system call with a pointer to the object code, such that calling the system call causes the object code to execute.

68. (Original) The computer program product of claim 67 further comprising:

program code for loading an interception module into computer memory, the interception module comprising the object code.

69. (Original) The computer program product of claim 67 further comprising:

program code for storing at least one association between the child processes and a network address.

70. (Original) The computer program product of claim 58 further comprising:

program code for detecting termination of a selected process; and

deleting all associations between the process and network addresses.

71. (Original) The computer program product of claim 70 further comprising:

program code for detecting termination of a selected process by intercepting system calls that terminate processes.

72. (Original) The computer program product of claim 71 further comprising:

program code for storing object code that deletes all associations between a selected process and network addresses; and

program code for replacing a pointer to a system call with a pointer to the object code, such that calling the system call causes the object code to execute.

73. (Original) The computer program product of claim 72 further comprising:

program code for loading an interception module into computer memory, the interception module comprising the object code.

74. (Original) The computer program product of claim 71 further comprising:

program code for deleting all associations between a selected process and network addresses.

75. (Original) The computer program product of claim 58 further comprising:

program code for, in response to a determination that the attempted communication is not via an associated network address, generating an error condition.

76. (Original) The computer program product of claim 75 further comprising:

program code for, in response to generating an error condition, not allowing the communication to proceed.

77. (Original) A computer program product for restricting network address-based communication by selected processes to a set of specific network addresses, the computer program product comprising:

program code for associating at least one selected process with at least one network address;

program code for determining whether an attempted network address-based communication of a selected process is via an associated address;

program code for, in response to a determination that the communication is not via an associated address, not allowing the attempted communication to proceed; and

a computer readable medium on which the program codes are stored.

78. (Original) A computer program product for restricting network address-based communication by selected processes to specific network addresses, the computer program product comprising:

program code for associating at least one selected process with at least one network address;

program code for detecting an attempt by a selected processes to associate a communication channel with a network address;

program code for determining whether the network address with which the selected process is attempting to associate a communication channel is associated with the selected process; and

a computer readable medium on which the program codes are stored.

79. (Original) The computer program product of claim 78 further comprising:

program code for, in response to a determination that the network address is associated with the selected process, allowing the communication channel to be associated with the network address.

80. (Original) The computer program product of claim 78 further comprising:

program code for, in response to a determination that the network address is not associated with the selected process, not allowing the communication channel to be associated with the network address.

81. (Original) A computer program product for restricting network address-based communication by selected processes to specific network addresses, the computer program product comprising:

program code for associating at least one selected process with at least one network address;

program code for detecting an attempt by a selected processes to associate a communication channel with a network address, wherein a provided value for the network address comprises a wild card;

program code for associating the communication channel with a network address that is associated with the process; and

a computer readable medium on which the program codes are stored.

82. (Original) The computer program product of claim 81 further comprising:

program code for associating the communication channel with a single network address with which the selected process is associated.

83. (Original) The computer program product of claim 81 wherein the selected process is associated with multiple network addresses; the computer program product further comprising:

program code for associating the communication channel with one of the multiple network addresses, resulting in a communication channel-network address pair;

program code for establishing one communication channel per each additional one of the multiple network addresses;

program code for associating each established communication channel with one of the multiple network addresses, resulting in additional communication channel-network address pairs; and

program code for associating the communication channel with the communication channel, network address pairs.

84. (Original) A computer program product for restricting network address-based communication by selected processes to specific network addresses, the computer program product comprising:

program code for associating at least one selected process with a unique local host address;

program code for detecting an attempt by a selected process to communicate with a local host;

program code for designating the unique local host address associated with the selected process to be used by the selected process to communicate with the local host; and

a computer readable medium on which the program codes are stored.

85. (Withdrawn) A computer program product for restricting network address-based communication by selected processes to specific network addresses, the computer program product comprising:

program code for associating at least one selected process with at least one network address;

program code for detecting an attempt by a selected processes to communicate with a second process via a communication channel;

program code for determining if the communication channel is associated with a network address;

program code for, in response to determining that the communication channel is not associated with a network address, associating the communication channel with a network address that is associated with the process; and

a computer readable medium on which the program codes are stored.

86. (Withdrawn) The computer program product of claim 85 further comprising:

program code for, in response to a determination that the communication channel is associated with a network address that is associated with the selected process, allowing subsequent communication via the communication channel.

87. (Withdrawn) The computer program product of claim 85 further comprising:

program code for, in response to a determination that the communication channel is associated with a network address that is not associated with the selected process, not allowing subsequent communication via the communication channel.

88. (Withdrawn) A computer program product for restricting network address-based communication by selected processes to specific network addresses, the computer program product comprising:

program code for associating at least one selected process with at least one network address;

program code for detecting an attempt by a selected processes to establish a connection between a communication channel and a second process;

program code for determining if the communication channel is associated with a network address;

program code for, in response to determining that the communication channel is not associated with a network address, associating the communication channel with a network address that is associated with the selected process; and

a computer readable medium on which the program codes are stored.

89. (Withdrawn) The computer program product of claim 88 further comprising:

program code for, in response to a determination that the communication channel is associated with a network address that is associated with the selected process, allowing the connection to be established.

90. (Withdrawn) The computer program product of claim 88 further comprising:

program code for, in response to a determination that the communication channel is associated with a network address that is not associated with the selected process, not allowing the connection to be established.

91. (Cancelled)

92. (Original) A method in a computer system for restricting network address-based communication by selected processes to a set of specific network addresses, the method comprising:

associating at least one selected process with at least one network address;

detecting when a selected process attempts to communicate via an unassociated address;

not allowing the attempted communication to proceed.

93. (Original) A computer program product for restricting network address-based communication by selected processes to a set of specific network addresses, the computer program product comprising:

program code for associating at least one selected process with at least one network address;

program code for detecting when a selected process attempts to communicate via an unassociated address;

program code for not allowing the attempted communication to proceed; and

a computer readable medium on which the program codes are stored.

94.-95. (Cancelled)